

Day 2. Todos: 第一个静态页面应用

作者: 王琦

创建日期: 2017 年 5 月 2 日

最后修改: 2017 年 5 月 8 日

今天, 你将使用一款被各种框架广泛实践的 **Todo List** 应用, 我们推荐你关注它的 **no freamework** 实现 (JQuery 实现)。在理解业务逻辑、代码实现后, 你将协同一位队友 (同样处在训练计划中的同学), 使用 **git** 共同发开一款属于你们的静态页面应用。

1. 使用 TodoMVC

下载项目:

```
git clone git@github.com:tastejs/todomvc.git
```

双击打开 ``todomvc/examples/jquery/index.html``, 体验这个静态页面应用。

- 因为 **jquery** 较为简单、广泛应用, 我们推荐以 ``jquery/js/app.js`` 为例, 理解这个应用是如何实现的。对于有一定 **JavaScript** 基础的同学, 也可以阅读其他实现中的 **JavaScript** 源码

注意到, 由于实现中使用的 ``localStorage`` 变量, 除非手动删除浏览器中的 ``localStorage``, 否则即使在浏览器中刷新这个页面, 此前添加的 **item** 将仍然存在。

2. 开发你自己的 Todo

2.1 任务

使用 **JavaScript** 与 **HTML**, 实现一个 **Todo List** 静态页面应用, 这个应用应包含的功能有:

- 添加一个 `活跃的(active)` **item** (类似的, 可以用回车键表示确认; 或者增添一个添加按钮)
- 将某个 **item** 标记为 `活跃的(active)` 或 `完成的(completed)`
- 查看某种类别的 **item**
 - 查看 `活跃的(active)` **items**
 - 查看 `所有的(all)` **items**
 - 查看 `完成的(completed)` **items**
- 删除 **item**
 - 删除某个 **item**
 - 批量删除所有已完成的 **items**

对于 **JavaScript** 的初学者, 我们推荐使用 **JQuery library** 来实现这个应用; 对 **JavaScript** 有一定了解的同学, 也可以选择自己感兴趣的 **framework** 或原生 **JavaScript**. 同时, 你也可以采用 **CSS3 (e.g. boost library)** 丰富界面。

2.2 工作流程

****1)** 在项目主页上记录分工******

在浏览器内打开项目主页 ``xx.xx.xx.xx/gitlab/your_group_name/your_project_name``, 在 ``issues`` 页面记录大家的工作内容以及应完成的时间 (建议精确到小时)

****2) 进行开发****

基于 `dev` 分支创建新的分支（建议分支名与你所负责的功能相关），进行开发。

****3) 发起并处理“合并请求” (Merge Request)****

在完成开发后，将代码 `push` 到远程。之后，在项目主页的 ``Merge Requests`` 上向你的队友 ``B`` 发起一个由 ``当前分支`` 向 ``dev`` 分支的合并请求。``B`` 在完成代码审查后，可以批注应该修改的内容，建议你重新修改；或者接受请求，合并这份代码。